WHAT IS CLAIMED IS:

1. A data processing system comprising:

a bus system;

a CPU connected to the bus system;

a RAM connected to the bus system, the RAM being divided into pages, each page having an execution flag;

a memory manager configured to manage the pages of the RAM and permit CPU execution of data on pages according to the execution flag;

a program stored within at least one page of the RAM; and

a program stack stored within at least one page the RAM,

wherein the memory manager is configured to determine whether the program is susceptible to buffer overflow attacks, and, if so, set the execution flag for program stack pages of RAM to deny CPU execution of data on the program stack pages of RAM.

2. The data processing system of claim 1 wherein the memory manager and the CPU are configured to deny CPU execution of data by triggering a hardware interrupt.

3. The data processing system of claim 1 further comprising:

a process structure table in data communication with the memory manager,

wherein the memory manager comprises an annotation API,

wherein the annotation API is configured to annotate within the process structure table the susceptibility of the program to buffer overflow attacks, and

wherein the memory manager is configured to make the determination of susceptibility to buffer overflow attacks with reference to the process structure table.

4. The data processing system of claim 2 further comprising:

a process structure table in data communication with the memory manager,

wherein the memory manager comprises an annotation API,

wherein the annotation API is configured to annotate within the process structure table the susceptibility of the program to buffer overflow attacks, and

wherein the memory manager is configured to make the determination of susceptibility to buffer overflow attacks with reference to the process structure table.

5. The data processing system of claim 1 further comprising:

a process structure table in data communication with the memory manager, and

an annotation program in data communication with the process structure table,

wherein the annotation program is configured to annotate within the process structure table the susceptibility of the program to buffer overflow attacks, and

wherein the memory manager is configured to make the determination of susceptibility to buffer overflow attacks with reference to the process structure table.

6. The data processing system of claim 3 wherein the program is configured to call the annotation API if the program is susceptible to buffer overflow attacks,

the memory manager is configured to determine susceptibility upon a request to allocate an additional page of RAM for the program.

7. The data processing system of claim 4 wherein the program is configured to call the annotation API if the program is susceptible to buffer overflow attacks,

the memory manager is configured to determine susceptibility upon a request to allocate an additional page of RAM for the program.

8. The data processing system of claim 5 wherein the program is configured to call the annotation program if the program is susceptible to buffer overflow attacks,

the memory manager is configured to determine susceptibility upon a request to allocate an additional page of RAM for the program.

9. A computer program product in a computer-readable medium adapted to prevent buffer overflow attacks comprising:

a memory manager code comprising a set of codes operable to direct a data processing system to manage a set of pages within a RAM of the data processing system and to permit a CPU of the data processing system to execute data on pages according to an execution flag on each of the set of pages;

an application program code comprising a set of codes operable to direct a data processing system to request the memory manager code to establish a program stack within at least one page the RAM; and

a susceptibility code comprising a set of codes operable to direct a data processing system to determine whether the application program code is susceptible to buffer overflow attacks, and, if so, set the execution flag for the program stack pages to deny CPU execution of data on the program stack pages.

10. The computer program product of claim 9 wherein the memory manager code further comprises a set of codes operable to direct a data processing system to deny CPU execution of data by triggering a hardware interrupt.

11. The computer program product of claim 9 further comprising:

a process structure table code comprising a set of codes operable to direct a data processing system to establish and maintain a process structure table code in data communication with the memory manager code and to annotate within the process structure table the susceptibility of the application program code to buffer overflow attacks,

wherein the memory manager code further comprises codes operable to direct a data processing system to make the determination of susceptibility to buffer overflow attacks with reference to the process structure table.

12. The computer program product of claim 10 further comprising:

a process structure table code comprising a set of codes operable to direct a data processing system to establish and maintain a process structure table code in data communication with the memory manager code and to annotate within the process structure table the susceptibility of the application program code to buffer overflow attacks,

wherein the memory manager code further comprises codes operable to direct a data processing system to make the determination of susceptibility to buffer overflow attacks with reference to the process structure table.

13. The computer program product of claim 11 wherein the memory manager code comprises the process structure table code as an API.

14. The computer program product of claim 12 wherein the memory manager code comprises the process structure table code as an API.

15. The computer program product of claim 11 wherein the application program code further comprises a set of codes operable to direct a data processing system to call the process structure table code the application program code is susceptible to buffer overflow attacks, and

the memory manager code further comprises a set of codes operable to direct a data processing system to determine susceptibility upon receipt of a request to allocate an additional page of RAM for the application program code.

16. The computer program product of claim 14 wherein the application program code further comprises a set of codes operable to direct a data processing system to call the process structure table code the application program code is susceptible to buffer overflow attacks, and

the memory manager code further comprises a set of codes operable to direct a data processing system to determine susceptibility upon receipt of a request to allocate an additional page of RAM for the application program code.

17. The computer program product of claim 13 wherein the application program code further comprises a set of codes operable to direct a data processing system to call the process structure table code the application program code is susceptible to buffer overflow attacks, and

the memory manager code further comprises a set of codes operable to direct a data processing system to determine susceptibility upon receipt of a request to allocate an additional page of RAM for the application program code.

18. A method for handling buffer overflow attacks against an application program running on a data processing system, having a CPU and a RAM divided into pages, the method comprising the steps of:

designating an execution flag for each page of RAM allocated to a stack of the application program;

permitting CPU execution of data on pages of RAM according to the execution flag;

determining whether the application program is susceptible to buffer overflow attacks; and

if the application program is susceptible to buffer overflow attacks, setting the execution flag as to the pages of RAM allocated to the stack of the application program to prohibit execution.

19. The method of claim 18 wherein step of permitting CPU execution comprises the steps of:

examining the execution flag on the page of RAM;

if the execution flag is not set, triggering a hardware interrupt; and

otherwise executing the data on the page.

20. The method of claim 18 further comprising the steps of:

establishing a process structure table;

maintaining the process structure table by annotating within the process structure table the susceptibility of the application program to buffer overflow attacks,

wherein the step of determining susceptibility to buffer overflow attacks is made with reference to the process structure table.

21. The method of claim 19 further comprising the steps of:

establishing a process structure table;

maintaining the process structure table by annotating within the process structure table the susceptibility of the application program to buffer overflow attacks,

wherein the step of determining susceptibility to buffer overflow attacks is made with reference to the process structure table.

22. The method according to claim 20 wherein the step of maintaining the process structure table is done once at the beginning of execution of the application program, and

the step of determining susceptibility is performed upon each receipt of a request to allocate an additional page of RAM for the application program code.

23. The method according to claim 21 wherein the step of maintaining the process structure table is done once at the beginning of execution of the application program, and

the step of determining susceptibility is performed upon each receipt of a request to allocate an additional page of RAM for the application program code.

24. The method according to claim 22 wherein request to allocate an additional page of RAM is a request to allocate the additional page of RAM for the stack.

25. The method according to claim 23 wherein request to allocate an additional page of RAM is a request to allocate the additional page of RAM for the stack.